REMARKS

Reconsideration and allowance are requested.

The Examiner maintains the Jaggar rejection arguing a "broadest reasonable interpretation" of the claim phrase "a common storage order compensated encoding with a subset of program instructions of said second instruction set such that, after compensating for storage order differences, all bits are identical." Although the Examiner's interpretation is certainly broad, Applicant does not agree it is reasonable when interpreted by one of ordinary skill in the art in light of the specification. "[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application." *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005) (*en banc*). It is the use of the words in the context of the written description and customarily by those skilled in the relevant art that accurately reflects both the "ordinary" and the "customary" meaning of the terms in the claims. *Ferguson Beauregard/Logic Controls v. Mega Systems*, 350 F.3d 1327, 1338 (Fed. Cir. 2003).

In paragraph 37 of the Office Action, the Examiner appears to be ignoring the fact that the encoding is such that after compensating for <u>storage order</u> differences, all bits are identical. The Examiner appreciates that the Thumb instruction set has 16 bits and the ARM instruction set has 32 bits. A person of ordinary skill in the art would not reasonably say that "compensating for storage order differences" includes a system in which it is necessary to double the number of bits present. That adding of further bits is not a storage order compensation; instead, it is a translation and/or expansion.

Furthermore, Figures 5 and 6 of Jaggar show that at least some bits <u>change</u> value when they are mapped between the two instruction sets. In Figure 5, the bit fourth from the left hand end of the Thumb instruction is a "0," but it maps to a "1" in the corresponding ARM instruction. Looking at the same bit positions in Figure 6, a "0" in the Thumb instruction maps to a "1" in the ARM instruction. Even if one considered that these bits have had their storage order changed, they are certainly not identical after this reordering. The Thumb instruction has been re-encoded.

The Examiner seems to acknowledge that not all bits can simply be copied over from the Thumb instructions to ARM instructions by admitting that "a slight mapping (storage differences)" is required. The sentence starting on page 13, line 12 of the Office Action explains that "*after conversion*, a 16-bit instruction is bit wide identical for (sic) a 32-bit instruction..." (emphasis added). This language acknowledges that Jaggar is converting (translating) and that Jaggar does not provide a system in which the instruction encodings of the instructions sets produce instructions that are bitwise identical after a storage order compensation. "Identical" is not the same as "almost identical." Identical encodings after storage order compensation permits considerable simplification. For example, an existing coprocessor infrastructure can be reused by providing identical coprocessor instructions within the two instruction sets after storage order compensation. "Almost identical" does not achieve such existing coprocessor infrastructure reuse. Additional complexity is also required, for example, a large number of existing coprocessor designs would need to be modified in order to permit them to operate with the almost identical but differing instructions. Hence, there is an important difference between instructions after storage order compensation being identical and instructions being non-identical.

The independent claims are amended to further distinguish from Jaggar: the "subset of program instructions of said first instruction set" has "a common bit-length" as well as "a common storage order compensated encoding." A basis for the added language "a common bit-length" can, for example, be found in Figures 4 and 5 where an ARM coprocessor instruction is shown as comprising 32 bits and an enhanced Thumb coprocessor instruction is also shown as comprising 32-bits formed of a first half word and a second half word. In contrast to this relationship between a subset of program instructions of the first instruction set and a subset of program instructions of the second instruction set as specified in such an amended claim. Jaggar's Thumb instructions are 16-bit instructions and must be mapped/converted to 32-bit ARM instructions. The two clearly have different lengths, and thus, do not satisfy the "common bit-length" claim language.

The application is in condition for allowance. An early notice to that effect is respectfully requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: ______________________
John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100